

**IN THE UNITED STATES PATENT  
AND TRADEMARK OFFICE**

# 2

Reissue Application No. 09/512,592 )

United States Patent No. 5,806,063 )

Group Art Unit: 2771

Issued: September 8, 1998 )

Examiner: Wayne Amsbury

Applicant: Dickens-Soeder2000, LLC )

Attorney Docket No.: 2039-301

Attorney Docket No. )

**PRELIMINARY AMENDMENT**

**RECEIVED**

**FEB 23 2000**

**REEXAM UNIT**

Honorable Commissioner of Patents and Trademarks  
Washington, D.C. 20231

Dear Sir:

Applicant in the above referenced Reissue Application to reissue the above  
referenced United States Patent hereby submits this Preliminary Amendment.

**Remarks**

Claims 1-15 are in United States Patent No. 5,806,063, entitled DATA  
FORMATTING AND SORTING FOR DATES SPANNING THE TURN OF THE  
CENTURY, issued to Dickens on September 8, 1998 on an application filed on October  
3, 1996 ("the patent"), as issued. New claims 16-76 were added to the Reissue  
Application as filed. The patentability of claims 1-15, in light of the Ohms article and the  
Shaughnessy '118 patent has been addressed in the Response of the Patent Owner to the  
Commissioner Initiated Order for Reexamination, filed contemporaneously with the  
patent owner's Reissue Application and this Preliminary Amendment, and incorporated

herein by reference ("the Response"). The patentability of claims 1-15 in light of additional references, including those asserted by Anonymous Submitters in the Reexamination Proceeding to invalidate those claims, and the patentability of the newly submitted claims in light of all references will be addressed in these Remarks. The additional references submitted by the Anonymous Submitters are set forth in Appendix A to this Preliminary Amendment. The patent owner has, in addition, filed an Information Disclosure Statement with the Reissue Application.

References which have been submitted by the Anonymous Submitters, listed in Appendix A, in many cases are not prior art references to the patent. The patent owner, however, in an abundance of caution will discuss these supposed references, but in doing so does not concede that any such supposed reference is actually prior art to the claimed invention as recited in the issued claims or in any of the newly submitted claims. These submitted references will be referred to herein as references, despite the disclaimer of acknowledgement herein that any such material is in fact a prior art reference. Many of these references disclose windowing and in many cases the use of a ten-decade window. None of them discloses the claimed invention, however, nor do any of them, alone or in combination with any others, render the claimed invention obvious. Most, if not all, teach away from the claimed invention.

The obviousness test under §103 must be applied to the subject matter of the claimed invention as a whole. There is no "gist" or "heart" or "core" of the invention evaluated for obviousness purposes. It is necessary to consider all of the subject matter of the claimed invention. Hawes, §7.04 at 7-6, citing, *Loctite Corp. v. Ultraseal Ltd.*, 781



§1.107(b). Hawes, §7.08, at 7-19, *citing*, *In re Newell*, 13 U.S.P.Q.2d 1248 (Fed. Cir. 1989), and *In re Kaplan*, 229 U.S.P.Q. 678, 683 (Fed. Cir. 1986).

A reference must be considered as a whole. A prior art reference that describes a product or process similar to the claimed product or process and also a statement that the product or process does not work does not teach the claimed product or process. A prior art reference may be considered to teach away when “ a person of ordinary skill, upon reading the reference, would be discouraged from following the path set out in the reference, or would be led in a direction divergent from the path that was taken by the [inventor].” Hawes, §7.05, at 7-7, *citing*, *In re Gurley*, 27 F.3d 551, 553, 31 U.S.P.Q.2d 1130, 1131 (Fed. Cir. 1994).

Simply because the claimed invention employs a known technique, i.e., windowing, does not, of itself, make the process of the claimed invention obvious. Hawes §7.05, at 7-8.1, *citing*, *In re Brower*, 77 F.3d 422, 37 U.S.P.Q.2d 1663 (Fed. Cir. 1994).

The claimed invention as recited in the original claims 1-15 has been discussed in the Response, and incorporated herein by reference. This discussion applies as well to claims 73 - 76, which are broader than the claims as originally issued. Claims 73 and 74 are identical respectively with claims 1 and 11, except that the recitation specifically of  $M_1M_2$  and  $D_1D_2$  have been omitted. The invention is broad enough to cover an alpha-numeric or other representation, e.g., a Julian representation, of the month and day. Claims 75 and 76 are identical, respectively, to claims 1 and 11 without the express recitation of a ten decade window. In regard to the claimed invention as recited in claims

16-72 the claims are more narrow than those of the original patent and serve to more fully and clearly define the claimed invention over the art.

The art does not teach or suggest “reformatting the symbolic representations of each of the symbolic representations of a date in the database, without the addition of any new data field to the database ... in order to facilitate collectively further processing the reformatted symbolic representations of each of the symbolic representations of each of the dates” as recited in new claims 16 and 26, and the claims that depend from these new claims. The same applies to new claims 31 and 32. The art does not teach or suggest “reformatting the symbolic representations of each of the symbolic representations of a date in the database, without changing any of the symbolic representations of a date in the database during the reformatting step ... in order to facilitate collectively further processing the reformatted symbolic representations of each of the dates” as recited in new claim 33 and the claims depending from that new claim. The art does not teach or suggest “converting each of the symbolic representations of dates stored in the at least one date field of the database to a symbolic representation of each of the respective dates that does not create the ambiguity, by windowing ... without the addition of any new data field to the database for purposes of such windowing and converting; and, running a program collectively on each of the converted symbolic representations of each of the respective dates ...” as recited in new claim 34 and the claims that depend from that new claim. The above comments also apply to new claims 60 and 61. New claims 62-65 include also a feature of “storing the converted symbolic representations separate from the at least one date field of the database” which is not taught or suggested by the art.

The art does not teach or suggest “reformatting the symbolic representations of each symbolic representation of a date in the at least one date field ... and repeating the step of reformatting until each symbolic representation of a date in the at least one date field has been reformatted ... in order to facilitate collectively further processing the reformatted symbolic representations ...” as recited in new claims 66 and 67. The art does not teach or suggest “reformatting ... in order to facilitate further processing of the reformatted symbolic representations of each of the symbolic representations of each of the dates, by running a program on the reformatted symbolic representations of each of the dates” as recited in new claim 68. The art does not teach or suggest “sorting the reformatted symbolic representations of the dates ... and, running a program on the reformatted symbolic representations of each of the dates” as recited in new claim 69. The art does not teach or suggest “converting each of the symbolic representations of dates stored in the at least one date field of the database ... without modifying any of the symbolic representations of dates in the at least one date field ...; and, running a program on the converted symbolic representations of each of the dates ... separately from the date data symbolic representations contained in the at least one date field of the database” as recited in new claim 70. The art does not teach “or suggest storing each of the converted symbolic representations of each of the dates separate from the database; and, running a program on the stored converted symbolic representations of each of the converted symbolic representations ...” as recited in new claim 71. The art does not teach or suggest “reformatting the symbolic representation of each symbolic representation of a date in the database ... prior to collectively further processing

information contained within the database associated with the respective dates” as recited in new claim 72.

Dependent claims 17-25 and 27-30 recite additional features that are patentable over the art for the reasons noted in the Response, incorporated herein by reference. Independent claims 35-59 add recitations to claims 33 and 34, respectively, that also are not taught or suggested by the art in combination with the recitations contained in independent claim 33 or 34, respectively.

The new claims are supported by the original specification and do not add any new matter. The Abstract states that “[d]ates stored in symbolic form in a database are reformatted to permit easy manipulation and sorting of [the] date-related information.” The Abstract also states that “[t]he reformatted date information is particularly useful ... because sorting by date is accomplished using a pure numerical-value sort.” The Specification further states that the “invention relates to the manipulation of information in a database, and in particular, to determination of dates in a useful form.” (Col. 1, lines 7-9) The Specification also notes that “[s]ets of dates spanning the turn of the century ... now stored in many databases ... may be manually converted to a more useable form in the sense that programs may be written to perform conversions, manipulations, and sorting. However, these programs typically require additional data fields for storage, which may be objectionable in some circumstances.” (Col. 1, lines 31-40) The Specification also notes that “there is a need for an improved approach to the representation and utilization of dates in databases, and for converting existing dates in databases to a more useable form.” (Col. 1, lines 41-45) Further notes the Specification, “an approach to the representation and utilization of dates stored symbolically in

09512597 011001  
00000000 00000000

databases [includes converting] [e]xisting symbolic representations ... to a more useful form without addition of new data fields, and in a manner that is performed automatically by the computer and requires no user input. The approach of the invention permits numerical sorting of dates.” (Col. 1, lines 51-56) The Specification also notes that “[a] century designator is determined for each date in the database [and] [e]ach date is formatted with the values  $C_1C_2 \dots$ .” (Col. 1, line 66 - Col. 2, line 3) The Specification also notes that “[t]his approach works particularly well if the dates are represented in the format  $C_1C_2 \dots$  [which] may be directly sorted numerically by fast sorting techniques, and thereafter stored back in the database.” (Col. 2, lines 15- 20) In addition, the Specification states “[t]he present invention thus provides an efficient approach to converting and utilizing symbolic date representations in databases, which allows automatic processing of dates ranging from before to after the year 2000. The large number of dates represented in some databases may thereby be readily processed and utilized.” (Col. 2, lines 21-26) The Specification also notes that “[t]his selection process [for the century designators] is performed in a completely automatic fashion by the computer, without human input other than to select the starting date of the 10-decade window.” (Col. 3, lines 36-38) Further, notes the Specification the “symbolic representations of the dates in the database are reformatted with the values  $C_1C_2 \dots$  . In one case that produces particularly advantageous results for many operations, such as chronological date sorting, the date is represented in the form  $C_1C_2 \dots \dots A$  straightforward numerical sort of date fields expressed in this form produces an accurate chronological ordering.” (Col. 3, lines 39-47) The Specification also notes that “[o]nce the symbolic representations of the dates are reformatted according to the procedures set



forth above, the date information may be sorted ... or otherwise manipulated, ... together with the dates, storing the dates and associated information back in the data base, or other processes.” (Col. 3, lines 48-55)

In regard to the references cited by Anonymous Submitter 1:

A SAS discloses “YEARCUTOFF+” as a “first year of a 100-year span used as a default by various DATE and DATETIME informats and functions.” It discloses “the relationship between the 100-year span and the two centuries if YEARCUTOFF= 1950.” SQL discloses “RR date format elements ... similar to the YY date format element, but it provides additional flexibility for storing date values in other centuries.” Schinkel discloses using windowing to solve the problem that “[d]ate fields in data entry screens rarely allow you to enter the century ... .” Whitney discloses “[a]pproaches to fixing [Y2K] problems” including to “[c]hange code to use a date window with YY” but it does not say what to do with the date window, and even cautions against its use “but ...”.

Straley discloses windowing to “adjust the internal year mechanisms for Clipper ... necessary if there is a string of a date that is to be converted back into the date format [and] for those applications that require string manipulations on dates outside of the currently adjusted century.” Clipper 5.0 discloses “an environment command that determines the interpretation of date strings containing only two year digits.” The July 27 posting describes software of a vendor that is said to employ a technique called “I/O fencing” wherein “[c]onversion code is added to the I/O to expand the date from the old format to the new (the century could be inferred here).” It then suggests that the “an attempt to use it across the board on our large applications, however, would have a high probability of foundering ... .” It also discloses using windowing “when the program

encounters a two-digit year ... ." It then says that this approach has problems when there is "a need to sort data on two-digit years." It also suggests using "MANDATE2, which apparently does windowing to infer a century "for as many date calculations as possible ... ." The July 16 posting discloses a date standard in which "[f]or presentation and input, yy-mm-dd is interpreted" using windowing. The August 7 posting discloses utilizing "sliding dates - leave as two digits & have some logic determine that 60 thru 99 implies 1960 thru 1999 and 00 thru 59 implies 2000 thru 2059." It also notes that "[t]he big problem with this approach is the difficulty of using the data in sorts, etc." The August 8 posting discloses the same thing as the August 7 posting and notes "[t]he programs need to be changed, but the problem still persists. If you sort on dates, look out." The August 3 posting discloses using "procedural logic to determine the century, for example, if year <35 then century =20 , else century = 19." It also notes that "[p]robably the most desirable path is to convert all two digit years to four digit years. Interface programs will be required for systems that have not been changed."

None of these disclose or suggest the claimed invention as recited in claims 1-15 as issued originally in the patent. To the extent they disclose windowing they utilize it in the manner of the '118 patent, on-call to process at most two date data entries at a time, and note the attendant problems, e.g., in doing sorting, or they utilize windowing in the manner of Roth et al., United States Patent No. 5,878,422, entitled SYSTEM FOR VIRTUALLY CONVERTING DATA IN A FIELD BETWEEN FIRST AND SECOND FORMAT BY USING HOOK ROUTINES, issued on March 2, 1999 on a application filed on April 9, 1997, which was a continuation-in-part of an application dated April 9, 1996, and assigned to Viasoft, Inc. Roth et al. has an effective date later than the date of

invention of the claimed invention, as demonstrated in the prosecution of the patent and is, therefore, not a prior art reference. Roth et al., however, does disclose a system which “assumes control each time a converted program reads or writes a record of a data file that has not yet been converted [and converts] records during I/O operations [enabling] a converted program to access an unconverted data file as if it were converted.” (Col. 2, lines 59-64) The disclosed embodiments employ a form of on-call windowing in a technique of I/O fencing by “intercept[ing] input/output (I/O) read requests from the program [and] intercept[ing] input/output (I/O) write requests from the program, and utilizing the conversion routine ... .” (Col. 3, lines 56-61) One form of conversion routine is windowing. (Col. 11, lines 20- 26).

09543590 "011801  
000000 26507560  
FOOTNOTES

Anonymous Submitter 3 filed a Request for Reexamination after the Commissioner had already ordered Reexamination. The patent owner herein addresses the art cited by Anonymous Submitter 3 as if submitted in the Reexamination proceeding initiated by the Commissioner. The citations of Anonymous Submitter 3 include Ohms and '118, discussed in the Response. The additional citations of Anonymous Submitter 3 also do not teach or suggest the claimed invention. Browe teaches “[d]ate calculations for FOCUS files.” He notes that “[a]ll available dates are in the form MMDDYYYY, using a 4 digit year. This is done to avoid additional program logic to calculate dates between the 20<sup>th</sup> and 21<sup>st</sup> century.” He also discloses a process to “change a date to other than the current date [in order to] ‘recreate’ accounting reports for previous accounting periods or preview future reports without changing program logic.” In doing this Browe teaches that “[t]he reference date is typed in MMDDYY format and must refer to a date from 01/01/1950 to 12/31-2049 for this focexec to work correctly.” He also notes that in

setting this "REFDATE" "[i]f the year is between 50 and 99, the date is assumed to be in the 20<sup>th</sup> century (1900-1999) else the 21<sup>st</sup> century (2000-2099)." Japan 05-027947 teaches a "2000 AD correspondence utility module [that is] activated when specified by program (7) or a utility and is positioned as pre-processing for processing that handles the year." (Translation, at 4). The Millenium Journal teaches that "[t]he following approaches to dealing with dates are not recommended." One such approach involves one form of "[l]ogic-based century determination," i.e. windowing, which "is only recommended as a temporary measure." After describing windowing the Journal notes that "[s]orts require an exit, and different date fields may require a different frame. Again, every place where dates are used in calculations, comparisons or displays, it is necessary to add additional logic." Milam teaches a system, like Ohms that stores and manipulates dates and times as a single binary number which represents the number of seconds that have elapsed since January 1, 1970 and teaches ways to convert this number into representations, including "building date strings." (at 72) He teaches "tp\_date takes a date string plus a format indicator and calculates a date (date\_t) value." "If the century in the date string is omitted, to\_date uses [windowing] to determine the century." (at 72-73) Lysgaard "deals with the problems [omission of century dates] causes in EDP systems as we approach the year 2000." He teaches that "[i]f information as to the valid time interval is added to the programme - maybe just the start for the 100 year interval - the programmes will be able to handle all time calculation correctly." (at 515) He also notes that "[a]lready today, most date manipulations take place via standard routines and ... as we approach year 2000, the standard routine library should probably be extended ... ." He also notes that "[w]hen one chooses to change only the programmes it is

necessary to discover which operations in the programmes will go wrong in year 2000. Such an analysis is also a fine basis for judging which standard routines that should be offered ... ." One of the areas on the "problem list" includes "[s]orting of 2 digit years with usual utility programmes will result in erroneous chronological number order.

Correct number order is achieved as follows:

definition of sorting sequence for first digit in year (decade), for example 6789012345 for time interval 1960-2059. This method can only be used if the interval is at the start of a decade.

temporary addition of auxiliary fields stating the century and included in the sorting criteria.

a sorting routine which can sort with transposed zero point. If the transposition for a 2 digit number area is set at '60', the order of the sorting should be 60-99, 00-59.

Lysgaard also notes that "[a] special problem presents itself if one has a 6 digit date included in the primary key of a file organized index sequentially." (at 516-17).

Shaw teaches windowing, but not specifically how to employ windowing, i.e., he does not teach the claimed invention. SAA AD/Cycle Language Environment, Programming Guide teaches a system, like Ohms, in which the dates are stored and manipulated in Lilian format (at xvi) and a number of "callable services" (at 31) which include CEEDAYS which uses windowing, like Ohms, to convert date strings into Lilian even if the century designator is not present. (at 84). SAS Language: Reference teaches "date and time informat that instruct the SAS System to read data values into variables that represent dates, times, and datetimes." (at 62) Similar to Ohms, SAS stores and

manipulates dates as a number representing the number of days since January 1, 1960. (at 85, 129)

None of the citations of Anonymous Submitter 3 teach or suggest the claimed invention as recited in claims 1-15 as originally issued in the patent. Neither do any of them teach or suggest the more narrowly defined claimed invention as recited in new claims 16-72 or the broadened claims 73-76.

Other prior art patents than those discussed above, under §102 (b), or under §102 (e), with effective dates before the date of the invention of the claimed invention, also do not teach or suggest the claimed invention. United States Patent No. 5,471,631, entitled USING TIME STAMPS TO CORRELATE DATA PROCESSING EVENT TIMES IN CONNECTED DATA PROCESSING UNITS, issued to Beardsley et al. on November 28, 1995 on an application dated October 19, 1992 teaches “a host processor(s) [that] is loosely-coupled by a plurality of data channels to a peripheral subsystem(s). The host processor(s) has a time of day clock. The peripheral subsystem(s) has a cluster(s) that performs peripheral controller functions. Each cluster has first and second clocks that respectively generate a log entry (logging) reference time and a subsystem time. The first and second clock times result in subsystem time stamps in a log that are not related to the time of day clock. For time correlating the time of day clock with the first and second clocks, a host time stamp is generated from the time of day clock.” (Abstract) United States Patent No. 5,526,515, entitled HARDWARE-DRIVEN CLOCK WITH OBJECT-ORIENTED INTERFACE, issued to Ross et al. on June 11, 1996, on an application dated June 30, 1994 teaches using “a floating point number ... for the total tick count to represent time so that both a large range and a high resolution can be accommodated,” as

shown in Table I. (Col. 3, lines 29-31) United States Patent No. 5,680, 622, entitled SYSTEM AND METHODS FOR QUICKLY DETECTING SHAREABILITY OF SYMBOL AND TYPE INFORMATION IN HEADER FILES, issued to Even on October 21, 1997 on an application filed on June 30, 1994 discloses a computer system which "eliminat[es] the storage of redundant data which is routinely present in parsed information." (Col. 3, lines 25-26) This is done by storing information in a "parse tree." (Col. 3, line 37) United States Patent No. 5,513,351, entitled PROTECTING A SYSTEM DURING SYSTEM MAINTENANCE BY USEAGE OF TEMPORARY FILENAMES IN AN ALIAS TABLE, issued to Grantz on April 30, 1996 on an application filed on July 28, 1994, discloses a "method of maintaining functional compatibility after performing system maintenance on a computer during system operation. The method ... determin[es] if a directory holding a set of files to be maintained has adequate space for a set of new files [and] in response to there being adequate space for the new set of files, places the new set of files in the directory. ... The new names of the new set of files are then placed in an alias table for access by any other files open on the computer during maintenance." (Abstract) United States Patent No. 5,513,306, entitled TEMPORAL EVENT VIEWING AND EDITING SYSTEM, issued to Mills et al. on April 30, 1996 on an application filed on June 7, 1995, which was a continuation of an application filed on April 11, 1994, which was a continuation of an application filed on August 9, 1990, discloses "a system for displaying a sequential stream of information on a single display." (Abstract)

A number of other patents having effective dates prior to the filing date of the patent, but not before the date of invention, address the Y2K problem in ways that also do

not teach or suggest the claimed invention. United States Patent No. 5,838,979, entitled PROCESS AND TOOL FOR SCALABLE AUTOMATED DATA FIELD REPLACEMENT, issued to Hart et al. on November 17, 1998 on an application filed on October 31, 1995, discloses "global modification of complex collections of interacting computer programs" among other things, for purposes of addressing the problem that "once these programs [which assume "that the two least significant digits of a year were adequate {sic} indicate the year"] begin to address years 2000 and beyond, there will be errors." (Col. 1, lines 44-51) Hart et al. teaches using "a global attribute database ... with the attributes of all variables which are subroutine parameters or arguments." Col. 32, lines 34-36) Also Hart et al. teaches "translation of the old program into an intermediate language." (Col. 2, lines 42-43) Hart et al. also teaches "expanding the year field to four digits," and "the addition of the logic for a century indicator where only a two-digit year representation is used and the user chooses to add the century indicator as a solution to the year representation problem." (Col. 10, lines 48-51) United States Patent No. 5,600,836, entitled SYSTEM AND METHOD FOR PROCESSING DATE-DEPENDENT INFORMATION WHICH SPANS ONE OR TWO CENTURIES, issued to Alter on February 4, 1997 on an application filed on November 14, 1995 discloses a "system that processes date-dependant information in which dates are specified in a two-digit format and are in at most two centuries. The system ... includes a number of local data files having dates represented in local time, a number of zone data files having dates represented in zone time, and at least one time change interface between local date files and zone date files that converts the dates represented in local time to dates represented in zone time and converts the dates represented in zone time to dates represented in local



time.” (Col. 2, lines 8-18) The local time dates are reduced by an amount in forming the zone time dates such that all zone time dates are within a single century and processing is done on them as if they were within a single century. (Col. 5, line 41-67) United States Patent No. 5,761,668 entitled METHOD AND APPARATUS FOR CONVERTING COMPUTER SOFTWARE AND DATABASES FOR THE YEAR 2000, issued to Adamchick on June 2, 1998 on an application dated October 29, 1996, claiming priority of a provisional application dated March 8, 1996, teaches a “method of modifying existing computer software and databases to utilize a date representation which accommodates dates subsequent to December 1, 1999 but preserves the desirable properties of utilizing no more than six numeric characters [with the] date ... represented as CYYDDD ... .” (Abstract) Adamchick teaches that windowing is a way to convert dates for the Y2K solution, but that Adamchick’s method is different. (Col. 3, lines 53-67) United States Patent No. 5,878,422, entitled SYSTEM FOR VIRTUALLY CONVERTING DATA IN A FIELD BETWEEN FIRST AND SECOND FORMAT BY USING HOOK ROUTINES, issued to Roth et al. on March 2, 1999, on an application filed on April 9, 1997, which was a continuation-in-part of an application filed on April 9, 1996, is discussed above. United States Patent No. 5,719,826, entitled CALENDARING SYSTEM, issued to Lips on February 17, 1998 on an application filed on May 1, 1996 teaches “[a] calendaring system which provides accurate calendaring of all dates between Jan. 01, 0001 and Dec. 31, 9999 by implementation of an extended set of leap-year rules.” (Abstract) The system stores and manipulates days in integer absolute days starting with January 1, 0001. (Col. 4, lines 38-50) The system “maintains consistency between two basic types of functions ... comput[ing] the number of days

between to dates [and] calculat[ing] a date based upon a starting date and a numerical increment.” (Col. 4, line 63 - Col. 5, line 2) A “parameter CENTURY-BREAK allows the user to tell the system how to assign a value to the century when the supplied date does not explicitly contain a century [using windowing].” (Col. 7, lines 1-12) United States Patent No. 5,644,762, entitled METHOD AND APPARATUS FOR RECORDING AND READING DATE DATA HAVING COEXISTING FORMATS, issued to Soeder on July 1, 1997 on an application filed on May 14, 1996, discloses a system for “using at least one of these unused ranges to integrate at least two formats for encoding the year data . . . . For new date data, i.e., after Dec. 31, 1999, the integer value of the year number is represented in binary format, and the binary representation is written in the two bytes already allocated for the year.” (Col. 2, lines 8-14) United States Patent No. 5,758,336, entitled DATE FORMAT AND DATE CONVERSION PROCEDURE USING A PACKED BINARY FORMAT, issued to Brady on May 26, 1998 on an application dated May 30, 1996, teaches “[a] binary date format, . . . ‘packed binary’, . . . compris[ing] a 2-byte data field. The first two bits of the data field are utilized as identifiers to enable the packed binary format to be identified. The remaining bits are employed as a binary value year indication. A date conversion procedure employs the packed binary format and is called when a program reaches a date field that is to be processed. The date conversion procedure initially examines the date field to determine whether its format is zone decimal, packed decimal or packed binary. If the date field data format is either zone decimal or packed decimal, the date values are converted to packed decimal and the required arithmetic procedure is performed, using the packed binary format date value. Once the calculation is complete, the procedure determines whether a “year 2000” switch

is set--indicating that all results are to be returned to the program in packed binary format. If the 2000 switch is not set, the date conversion procedure reconverts the packed binary year format back to the originally received format and returns the result to the running program.” (Abstract) United States Patent No. 5,970,247, entitled METHODS FOR ENCODING DECODING AND PROCESSING SIX CHARACTER DATE DESIGNATIONS FOR THE YEAR 2000 AND BEYOND, issued to Wolf on October 19, 1999 on an application filed on October 7, 1997, claiming priority from a number of provisional applications going back to June 17, 1996, discloses “methods for encoding (and decoding) calendar dates in multiple centuries, as sequences of six characters.” (Col. 2, lines 64-67) Wolf teaches the use of “non-numeric characters, or of four-digit numbers greater than 1231, to represent the month and day in dates in and after a specified year, e.g., the year 2000.” United States Patent No. 5,808,889, entitled SYSTEM AND METHOD FOR IDENTIFYING AND CORRECTING COMPUTER OPERATIONS INVOLVING TWO DIGIT YEAR DATES, issued to Burgess on September 15, 1998 on an application dated June 28, 1996, teaches a “system and method for identifying and correcting computer operations involving two digit year dates. The system includes: a computer and a routine for searching for a subtraction, comparison and sort operations on the computer involving a plurality of two digit quantities representing year dates. The subtraction operation is executed to obtain a difference value between the two quantities. A negative difference value is verified and a plurality of two digit numbers whose sum is 100 is added to the negative difference value so that the subtraction operation yields a correct result for two digit quantities representing years greater than or equal to the year 2000. A similar routine is added to existing computer code involving compare and/or

sort operations.” (Abstract) United States Patent No. 5,740,442, entitled METHOD AND APPARATUS FOR IDENTIFYING AND CORRECTING DATE CALCULATION ERRORS CAUSED BY TRUNCATED YEAR VALUES, issued to Cox, et al. on April 14, 1998 on an application filed on August 27, 1996 teaches a “standardized test environment [that] assists identifying problematic instances of instructions in computer programs that produce erroneous results caused by dates having years expressed in a truncated form. ... In one embodiment of a production environment, a computer system intercepts execution of a computer program at specified locations and takes corrective action according to information obtained from a table of control information.” (Abstract)

United States Patent No. 5,794,048, entitled METHOD FOR CLASSIFICATION OF YEAR-RELATED DATA FIELDS IN A PROGRAM, issued to Brady on August 11, 1998, on an application filed on August 29, 1996, teaches “enabl[ing] a computer to examine a software application composed of one or more programs, which includes operand labels and logical or computational operators, and to identify operand fields which include a year value. ... Thereafter, the method determines, for operator entries located in the operator table, and from operands associated therewith and an operand association table, whether the classification for each respective operand in the operand table should or should not be altered.” (Col. 2, lines 8-28) Brady also teaches that this procedure “enables identification of year fields with high probability, and thereafter enables each of these fields to be automatically accessed and to have their year designation altered in such a manner as to enable a removal of the ambiguity which arises from the onset of the year 2000 and forward. A preferred method for revising the year format for enabling the year conversion procedure is described in U.S. patent application

Serial No. 08/657,657 ... entitled 'Date Format and Date Conversion Procedure' to Brady." (Col. 9, line 63 - Col. 10, line 5) United States Patent No. 5,956,510, entitled APARATUS AND METHOD FOR REVISING COMPUTER PROGRAM CODE, issued to Nicholas on September 21, 1999 on an application filed on September 10, 1996 teaches an approach to the Y2K problem that is asserted to be in lieu of an "attempt to expand a two-digit field to four digits," (Col. 1, lines 37-38) or using "logic that assumes that the century has changed if the year value falls into a specified range." (Col. 1, lines 51-53) Nicholas teaches an "apparatus and methods for revising a field in computer program code [which] includes the steps of analyzing program code lines for the presence of the field therein and, if present, revising the field from the first value range to a second value range." (Col. 2, lines 13-17) Nicholas further teaches "inserting an extended field representative of a base value into the program code, with the field being encoded as an offset value relative to the extended field." (Col. 2, lines 21-21) United States Patent No. 5,668,989, entitled TWO-DIGIT HYBRID RADIX YEAR NUMBERS FOR YEAR 2000 AND BEYOND, issued to Mao on September 16, 1997 on an application filed on September 18, 1996, discloses the use of "numbers ... represented with 2 digit hybrid radix numbers, in which the higher digit is input, generated, processed, and output as hexadecimal, but displayed in a decimal-like way with font patterns 0-9 and '0-'5, while the lower digit is treated as ordinary decimal." (Col. 2, lines 37-41) In this manner "no changes to the existing application source code other than a re-compilation is needed, and compatibility with existing data files [sic files] and databases is automatically maintained. All data files need no change, and all the databases will continue to work

with a re-compiled DBMS while no change is needed for the database schema and contents.” (Col. 2, lines 44-50)

Important discussions of the Y2K problem around the time of the invention of the claimed invention also teach away from the claimed invention. These include the IBM publications, *The Year 2000 and 2-Digit Dates: a Guide for Planning and Implementation*, IBM, First Edition, October 1995 (“the Guide, First Ed.”), and *The Year 2000 and 2-Digit Dates: a Guide for Planning and Implementation*, IBM, Third Edition, May 1996 (“the Guide, Third Ed.”), each of which was of record and sworn behind in the initial prosecution of the patent, and J. Murray and M. Murray, *The Year 2000 Computing Crisis*, McGraw-Hill, 1996, Chapter 11, The Conversion Plan, (“Murray”), which was submitted anonymously in the Reexamination proceeding by Anonymous Submitter 2, and asserted by Anonymous Submitter 2 to have been published in March of 1996.

The Guide First Ed. does not teach or suggest the claimed invention. It notes in Chapter 5, at 5-1, which contains “a number of techniques ... to correct improper date notation and use” that the number one choice is “[c]onversion to [f]ull 4-[d]igit-[y]ear [f]ormat, [which] requires changes to both the data and the programs by converting all references and/or uses of 2-digit year format (YY) to 4-digit-year format (YYYY).” (*Id.* at 5-1)

The Guide First Ed. then suggests using a form of windowing (“Fixed Windowing Technique”) such that “[p]rogram logic determines the century based on the following data checking. If the 2-digit year representation of a specific year is *xy* then if:

*xy* ≥ 60, then it is a 20<sup>th</sup> century date (19*xy*)

otherwise (that is, *xy* ≤ 59), it is a 21<sup>st</sup> century date (20*xy*).”

This suggestion is to change dates in an on-call routine in the manner of the Shaughnessy '118 patent discussed in the Response. The Guide, First Ed. notes that in using this technique the user should "[e]xpect a performance impact in direct proportion to the quantity of date processing the particular application handles due to the overhead of 2- to 4- digit year conversion." The Guide First Ed. also notes that "[a]ll programs that use the fixed window technique may need to be manually updated on a yearly basis depending on how your date routine is packaged." In addition the Guide First Ed. notes "[a]ll programs that accept output from the fixed window technique must use the same assumptions (current date, past and future windows)."

Finally the Guide First Ed. notes that:

[r]etaining a 2-digit year representation does not provide collating sequence support. Nor does the use of a fixed window technique provide indexing sequence support when 2 digit years are used as index keys in indexed files. You need to provide additional processing to obtain correct collating and indexing sequence output.

The Guide, First Ed. also suggests the use of a "Sliding Window Technique."

Except for the "manually updated" requirement, the above comments are made with respect to the sliding windowing technique as well.

The Guide, First Ed. does not teach doing sorting utilizing the windowing technique disclosed in the Guide, First Ed. In either the Guide Second Edition or the Guide Third Ed., as appears in the Guide, Third Ed., there is added the following sentence to the description of the sliding windowing technique:

[f]or an example of how DFSORT/MVS is implementing a sliding window to sort, merge, and transform 2-digit year data to 4-digit year data, refer to “DFSORT” on page 7-23.

The section “DFSORT” as shown in the Guide, Third Ed., at 7-23, and the Guide, Ninth Edition, at 8-25, does not appear in the Guide, First Ed. The DFSORT discussion also indicates that the Guide, Third Ed. does not teach or suggest the claimed invention:

In Murray, the authors suggest a “conversion method” which utilizes windowing, but not in the manner of the claimed invention. It teaches the total conversion of all database date entries to four character date data, along with all of the programs. Indeed, it teaches that to “create a program environment whose logic is internally eight-digit-date-oriented while all of its output is six-digits dates[ i.e., to] patch programs without having to update databases” is **not** the appropriate approach. (*Id.* at 204) “Conversion to an eight-digit-date standard both within programs and within databases staves off ultimate disaster and preserves the investment demanded by the current [Y2K] crisis.”(*Id.*)

A number of patents which are clearly not part of the prior art have been issued by the United States Patent and Trademark Office which show various approaches to solving the Y2K problem. Many of these disclose or imply various uses of windowing in the contexts of the differing solutions taught and suggested in these patents. These patents include United States Patent No. 5,806,067, entitled METHOD AND APPARATUS FOR DATA PROCESSING WITH CONVERSION OF AMBIGUOUS DATA STRUCTURES, issued to Connor on September 8, 1998 on an application filed on October 21, 1996; United States Patent No. 5,845,286, entitled DATE VALUE



REDUCTION SYSTEM, issued to Colizza on December 1, 1998, on an application dated December 24, 1996; United States Patent No. 5,978,809, entitled, METHOD OF SOLVING MILLENNIUM PROBLEMS OF SOME APPLICATION PROGRAMS, issued to Berner on November 2, 1999 on an application filed on January 27, 1997; United States Patent No. 5,758,346, entitled CONVERTING REPRESENTATIONS OF YEAR, issued to Baird on May 26, 1998 on an application filed on January 29, 1997; United States Patent No. 5,809,500, entitled SYSTEM FOR CONVERTING PROGRAMS AND DATABASES TO CORRECT YEAR 2000 PROCESSING ERRORS, issued to Nolan on September 15, 1998, on an application filed on June 20, 1997, claiming priority from a provisional application filed on February 26, 1997; United States Patent No. 5,915,116, entitled TIME VALUE MANIPULATION, issued to Hochman et al. on June 22, 1999, on an application filed on March 7, 1997; United States Patent No. 6,009,540, entitled LOGIC MODULE FOR IMPLEMENTING SYSTEM CHANGES ON PC ARCHITECTURE COMPUTERS, issued to Craft et al. on December 28, 1999, on an application filed on April 8, 1997; United States Patent No. 5,903,895, entitled METHOD FOR REFORMATION CONVENTIONAL THREE FIELD DATE FORMATS TO PRODUCE A CENTURY ACCUMULATED DATE, issued to Hoffman et al. on May 11, 1999, on an application dated April 29, 1997; United States Patent No. 5,852,824, entitled APPARATUS AND METHOD FOR PROCESSING YEAR-DATE DATA IN COMPUTER SYSTEMS, issued to Brown on December 22, 1998 on an application filed on May 22, 1997; United States Patent No. 5,797,117, entitled MONTH FIELD DIVISION MULTIPLEXING SOLUTION FOR YEAR 2000 COMPUTER DATE PROBLEM, issued to Gregovich on August 18, 1998

on an application filed on May 24, 1997; United States Patent No. 5,802,354, issued to Kubala et al. on September 1, 1998, on an application filed on June 9, 1997, entitled METHOD AND APPARATUS FOR SYNCHRONIZING SELECTED LOGICAL PARTITIONS OF PARTITIONED INFORMATION HANDLING SYSTEM TO A TEST DATASOURCE; United States Patent No. 5,911,142, entitled SYSTEM AND METHOD FOR BRIDGING COMPLIANT AND NON-COMPLIANT FILES, issued to Smith et al. on June 8, 1999, on an application filed on August 5, 1997, claiming priority of a provisional application filed on July 1, 1997; United States Patent No. 6,002,873, entitled, COMPILER-ASSISTED OR INTERPRETER ASSISTED WINDOWING SOLUTION TO THE YEAR 2000 PROBLEM FOR COMPUTER PROGRAMS, issued to Carter et al. on September 14, 1999 on an application filed on November 14, 1997, which was a continuation in part of an application filed on July 23, 1997; United States Patent No. 5,765,145, entitled METHOD FOR PERFORMING DATE OPERATIONS ON DATE FIELDS SPANNING SEVERAL CENTURIES, issued to Masiello on June 9, 1998 on an application filed on July 28, 1997; United States Patent No. 5,926,814, entitled SYSTEM AND METHOD FOR PROCESSING A NEW CALENDAR SYSTEM, issued to Friedman on July 10, 1999, on an application filed on September 22, 1997; and United States Patent No. 6,003,028, entitled IMPLEMENTING EXTENDED NUMERIC RANGE WITHIN A TWO DIGIT SOFTWARE REPRESENTATION, issued to Koenig on December 14, 1999 on an application filed on January 13, 1998 which claimed priority from a provisional application filed on December 16, 1997.

For the above stated reasons, the applicant submits that the claimed invention as recited in claims 1-15 of the patent, as issued, and in the new claims 16-76 added in the

William C. Cray  
Levin & Hawes  
Attorneys for Dickens-Soeder  
February 22, 2000

27